

REMARKS

The Examiner rejected claims 1-13 under 3 U.S.C. §102(c) as allegedly being anticipated by Bowman-Amuah (US 6636424).

Applicants respectfully traverse the §102 rejections with the following arguments.

35 U.S.C. §102(e)**Claims 1-6**

The Examiner rejected claims 1-13 under 3 U.S.C. §102(e) as allegedly being anticipated by Bowman-Amuah (US 6636424).

Applicants respectfully contend that Bowman-Amuah does not anticipate claim 1, because Bowman-Amuah does not teach each and every feature of claim 1.

For example, Bowman-Amuah does not teach the feature "a handler sub-system including controller objects which control a sequence of actions by the business logic sub-system in a use case, in response to an event triggered by the view sub-system".

As another example, Bowman-Amuah does not teach the feature: "a view context sub-system including at least one context object which is arranged to capture input and output data which populates the presentation objects of the view sub-system".

Based on the preceding arguments, Applicants respectfully maintain that Bowman-Amuah does not anticipate claim 1, and that claim 1 is in condition for allowance. Since claims 2-6 depend from claim 1, Applicants contend that claims 2-6 are likewise in condition for allowance.

Claims 7-12

Applicants respectfully contend that Bowman-Amuah does not anticipate claim 7, because Bowman-Amuah does not teach each and every feature of claim 7.

For example, Bowman-Amuah does not teach the feature: "at least one handler object which controls actions of at least one of the view objects and actions of at least one of the business

objects". The Examiner's argument that the flowchart in FIG. 158 of Bowman-Amuah teaches the preceding feature of claim 7 is incorrect, because FIG. 158 of Bowman-Amuah does not disclose that the data handler controls actions of at least one of the view objects, and also because FIG. 158 of Bowman-Amuah does not disclose that the data handler controls actions of at least one of the business objects. In fact, FIG. 158 of Bowman-Amuah does not disclose anything at all about the actions of at least one of the view objects or of at least one of the business objects.

As another example, Bowman-Amuah does not teach the feature: "at least one view context object comprising data objects which capture a state of at least one of the view objects". The Examiner's argument that the col. 203, line 3 of Bowman-Amuah teaches the preceding feature of claim 7 is incorrect, because col. 203, line 3 of Bowman-Amuah does not disclose anything at all about the view objects.

Based on the preceding arguments, Applicants respectfully maintain that Bowman-Amuah does not anticipate claim 7, and that claim 7 is in condition for allowance. Since claims 8-12 depend from claim 7, Applicants contend that claims 8-12 are likewise in condition for allowance.

Claim 13

Applicants respectfully contend that Bowman-Amuah does not anticipate claim 13, because Bowman-Amuah does not teach each and every feature of claim 13.

As a first example of why Bowman-Amuah does not teach each and every feature of claim 13, Bowman-Amuah does not teach the feature: "creating a view object with the handler object". The Examiner argues that Bowman-Amuah, col. 119, line 66 recites: "*Interface logic interprets and maps the actions of users into business logic processing activities. With the assistance of*

Presentation Services, Interface logic provides the linkage that allows users to control the flow of processing within the application." In response, Applicants respectfully contend that the preceding argument by the Examiner is incorrect, because Bowman-Amuah, col. 119, line 66 does not disclose anything at all about the creation of view objects with a handler.

As a second example of why Bowman-Amuah does not teach each and every feature of claim 13, Bowman-Amuah does not teach the feature: "creating a view context object with the view object". The Examiner argues that the User Interface Component of FIG. 42 of Bowman-Amuah teaches the preceding feature of claim 13. In response, Applicants respectfully contend that there is nothing in FIG. 42 of Bowman-Amuah that discloses creating a view context object. In addition, FIG. 42 of Bowman-Amuah does not disclose the view object that has antecedent basis in the view object created by the handler object.

As a third example of why Bowman-Amuah does not teach each and every feature of claim 13, Bowman-Amuah does not teach the feature: "passing the view context object to the handler object". The Examiner argues that col. 275, line 56 of Bowman-Amuah recites: *"FIG. 158 illustrates a flowchart for a method 15800 for controlling data. A data retrieval mechanism is provided in operation 15802 for retrieving data from a database. The data retrieval mechanism also writes data to the database. In operation 15804, the data retrieval mechanism is encapsulated in a data handler. A request from a domain object is received for a retrieval of a portion of the data in the database in operation 15806. The data retrieval mechanism is utilized in operation 15808 to retrieve the portion of the data from the database. The portion of the data is passed through the data handler to the domain object in operation 15810."* In response, Applicants maintain that col. 275, line 56 of Bowman-Amuah does not disclose anything at all about the view

context object that has antecedent basis in the view context object that is created with the view object. In addition, Applicants maintain that col. 275, line 56 of Bowman-Amuah does not disclose anything at all about the handler object that has antecedent basis in the handler object creates the view object. In fact, col. 275, line 56 of Bowman-Amuah is totally irrelevant to the preceding feature of claim 13.

As a fourth example of why Bowman-Amuah does not teach each and every feature of claim 13, Bowman-Amuah does not teach the feature: "updating the view context object with the handler object". The Examiner argues that col. 102, line 33 of Bowman-Amuah recites: "*an application may be designed to retrieve the tax rate for the State of Illinois. When the user enters "Illinois" on the screen, the application first validates the user's entry by checking for its existence on the "State Tax Table" and then retrieves the tax rate for Illinois. Note that codes tables provide an additional degree of flexibility. If the tax rates changes, the data simply needs to be updated; no application logic needs to be modified.*" In response, Applicants maintain that col. 102, line 33 of Bowman-Amuah does not disclose anything at all about the view context object that has antecedent basis in the view context object that is created with the view object. In addition, Applicants maintain that col. 102, line 33 of Bowman-Amuah does not disclose anything at all about the handler object that has antecedent basis in the handler object creates the view object. In fact, col. 102, line 33 of Bowman-Amuah is totally irrelevant to the preceding feature of claim 13.

As a fifth example of why Bowman-Amuah does not teach each and every feature of claim 13, Bowman-Amuah does not teach the feature: "refreshing the view object from the updated view context object". The Examiner argues that col. 296, line 13 of Bowman-Amuah recites: "*A retrieveData() message might also be required, if the object model pre-instantiates objects before*

09/966,131

10

retrieving them. Similarly, refresh() could be used: the business object, if dirty, replaces any changes with data originally from the data store." In response, Applicants maintain that col. 296, line 13 of Bowman-Amuah does not disclose anything at all about the view object and the updated view context object.

Based on the preceding arguments, Applicants respectfully maintain that Bowman-Amuah does not anticipate claim 13, and that claim 13 is in condition for allowance.

CONCLUSION

Based on the preceding arguments, Applicants respectfully believe that all pending claims and the entire application meet the acceptance criteria for allowance and therefore request favorable action. If the Examiner believes that anything further would be helpful to place the application in better condition for allowance, Applicants invites the Examiner to contact Applicants' representative at the telephone number listed below. The Director is hereby authorized to charge and/or credit Deposit Account No. 09-0457.

Date: 10/27/2004

Jack P. Friedman
Jack P. Friedman
Registration No. 44,688

Schmeiser, Olsen & Watts
3 Lear Jet Lane, Suite 201
Latham, New York 12110
(518) 220-1850